AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended) A program control apparatus for controlling execution of a program in a computer system in which a plurality of threads are switched in accordance with a scheduling policy by a scheduler, comprising:

first means responsive to a predetermined first application program interface call from one of said plurality of threads for setting a prescribed flag to one of first and second states,[[;]] said first means includes means responsive to an application program interface call from one of said plurality of threads which requests start of detection of presence/absence of a context switching, for setting a flag indicating presence/absence of a context switching to a state indicating absence of a context switching, and for disabling context switching between threads when said flag is set to correspond to absence of context switching;

second means for detecting, after said flag is set to said one state, a prescribed change in a state of said scheduling policy, and for setting said flag to a other one of said first and second states,[[; and]] said second means includes means for setting, after said flag is set to the state corresponding to the absence of a context switching, said flag to a state corresponding to presence of a context switching, and for enabling context switching between threads when said flag is set to correspond to the presence of context switching; and

third means responsive to a predetermined second application program interface call constituting a pair with said first application program interface, from said thread, for returning a value indicative of the state of said flag to said thread,[[.]] said third means

includes means responsive to an application program interface call from one of said plurality of threads which requests termination of detection of presence/absence of a context switching, for returning a value corresponding to the state of said flag to said thread.

Claim 2 (cancelled)

Claim 3 (original) The program control apparatus according to claim [[2]]1, further comprising means for invalidating, when there is said context switching, processing of said thread from said application program interface call requiring start of detection of presence/absence of a context switching until said application program interface call requesting termination of detection of presence/absence of a context switching.

Claim 4 (original) The program control apparatus according to claim 3, further comprising:

means for alternately changing priority of said thread to high and low; and

means receiving a process time from said application program interface call requesting start of detection of presence/absence of a context switching until said application program interface call requesting termination of detection of presence/absence of a context switching, for comparing the received process time with a remaining time until the priority of said thread is changed to low, when said thread is at a high priority state, and upon detection that said remaining time is shorter than said process time, lowering the priority of said thread.

Claim 5 (original) The program control apparatus according to claim 3, wherein said thread is a garbage collection thread in accordance with copy method in which an object which is referenced by any other object in a memory heap area is detected, and the object is copied to a prescribed area in said heap area.

Claim 6 (original) The program control apparatus according to claim 3, wherein said thread is a memory compaction thread for eliminating fragmentation, by freeing a memory area of an object not referenced by any other object in a memory heap area as a free memory area allocatable to other object.

Claim 7 (original) A program control apparatus according to claim 1, wherein

said first means includes means responsive to an application program interface call from a thread which interface request start of detection of presence/absence of a data write to a designated memory area, for setting a flag indicating presence/absence of a data write to a state corresponding to absence of a data write;

said second means includes means for setting, when there is a data write to said designated memory area, said flag to a state corresponding to presence of a data write; and setting said flag to another state when there is no data write to the designated memory area; and

said third means includes means responsive to an application program interface call from said thread which interface requests termination of detection of presence /absence of a data write to the designated memory area, for returning a value corresponding to the state of said flag to said thread.

Claims 8-23 (canceled)

Claim 24 (previously presented) A method of program control, comprising the steps of:

in response to an application program interface call from a thread from a plurality of threads which requests start of detection of a presence or absence of context switching, setting a flag indicating absence of context switching;

disabling context switching between threads when said flag is set to correspond to absence of context switching;

after said flag is set to the state corresponding to absence of context switching, setting said flag to a state corresponding to presence of context switching;

enabling context switching between threads when said flag is set to correspond to presence of context switching; and

in response to an application program interface call from said thread which interface requests termination of detection of such thread.

Claim 25 (previously presented) A method of program control, comprising the steps of:

in response to an application program interface call from a thread from a plurality of threads which requests start of detection of a presence or absence of a data write to a designated memory area, setting a flag indicating the absence of a data write;

setting said flag to a state corresponding to presence of a data write when there is a data write to said designated memory area[[;]] and setting said flag to another state when there

is no data write to the designated memory area; and

in response to an application program interface call from said thread which interfaced requests termination of detection of presence/absence of a data write to the designed memory area, returning a value corresponding to the state of said flag to said thread.

Claims 26-31 (canceled)

Claim 32 (previously presented) A computer readable recording medium storing a program control program allowing a computer to execute a program control method, said program control method includes the steps of

in response to an application program interface call from a thread which requests start of detection of a presence or absence of context switching, setting a flag indicating the absence of context switching;

disabling context switching between threads when said flag is set to correspond to absence of context switching;

in response to an application program interface call from the thread, and after said flag is set to the state corresponding to the absence of context switching, setting said flag to a state corresponding to presence of context switching; ~~and~~

enabling context switching between threads when said flag is set to correspond to presence of context switching; and

in response to an application program interface call from said thread which requests termination of detection of the presence or absence of a context switching, returning a value corresponding to the state of said flag to said thread.

Claim 33 (previously presented) A computer readable recording medium storing a program control program allowing a computer to execute a program control method, said program control method comprising the steps of:

in response to an application program interface call from a thread which requests start of detection of a presence or absence of a data write to a designated memory area, setting a flag indicating an absence of data writes;

setting said flag to a state corresponding to presence of a data write when there is a data write to said designated memory area[[;]] and setting said flag to another state when there is no data write to the designated memory area; and

in response to an application program interface call from said thread which interface requests termination of detection of presence/absence of a data write to the designated memory area, returning a value corresponding to the state of aid flag to said thread.

Claims 34-42 (canceled)